CSCI 5832 Final Project Report (Source Code) Lucas Derr May 8, 2024

Anonymous ACL submission

1 Abstract

Fine-tuning a large-language model has become a fundamental skill in natural language processing. These powerful models are malleable and can be fine-tuned to perform specific tasks. This project uses hip-hop lyrics scraped from the internet to fine-tune the distilgpt2 model (a distilled version of GPT-2). The newly trained model is capable of generating text and "auto-completing" user-inputted lyrics. This report will outline all the steps taken to achieve this final result, including data scraping/collection, data cleaning, model training, and evaluation. Finally, as promised, the final model (Rhymebot 1.0) has released an album using some prompts that I have given it. This is included in the appendix.

2 Introduction

Songwriting is a creative process that often requires inspiration and skill. Automating lyric generation could assist musicians who are stuck in a creative block. Additionally, writing lyrics could be considered a 'human' task due to its need for significant emotional understanding. It might be difficult or impossible for a mathematical framework such as a transformer to authentically capture the essence of the human experience. Despite this, one cannot ignore the powerful AI models which are still capable of lyric generation can mimic creativity and even novelty. While these types of models will never replace humans, they can be a helpful tool in enhancing the songwriting process.

Building such a model requires a few key steps. First, and most importantly, a dataset must be either generated or downloaded from the internet. To gain experience in web-scraping, a script was written to aggregate a large, customized hip-hop lyric dataset. After the data is gathered and correctly compiled, it is cleaned and prepped so it can be fed into the fine-tuning of the transformer model. After the data

is clean, it is used to retrain the model and adjust its weights. This project elected to use distilgpt2, as it is relatively lightweight and well-regarded in the NLP community. After the model has finished training, it can be evaluated with a combination of objective and subjective metrics. Each of these steps will be explained in greater detail in the Description of Methodology section of the report.

3 Related Work

3.1 GhostWriter: Using an LSTM for Automatic Rap Lyric Generation

https://aclanthology.org/D15-1221.pdf

This 2015 paper is very similar to what I will be doing. These researchers used an LSTM to try to generate rap lyrics in the style of a specific artist. Their main metric measures the similarity between the generated lyrics and rhyme density. They conclude that their LSTM model is more effective at generating lyrics than the baseline n-gram model. My project differs because I will use a pretrained transformer and my dataset will include many artists' lyrics.

3.2 Hierarchical attention based long short-term memory for Chinese lyric generation

This 2018 paper uses a hierarchical attention model to capture contextual information about the lyrics. The goal with this framework is to capture lyrical style and patterns, maintaining cohesion while generating lyrics. They use cross entropy loss and perplexity to evaluate their model and conclude that the model beats state-of-the-art models, but fails to live up to lyrics written by humans. My project is different because I will not be using a hierarchical attention framework.

3.3 Syllable-level lyrics generation from melody exploiting character-level language model

This 2024 paper focuses on training a large language lyric generator model at the syllable level in order to establish a mapping between musical notes and syllables. They elect to use ROUGE, BLEU, and BERT for evaluation but emphasize that there exists no metric that comprehensively captures the quality of generated lyrics. Additionally, they use a rating scale for subjective human evaluation. They conclude that the syllable-based model improves naturality, correctness, and coherence of lyrics'.

4 Lyrics Generation supported by Pre-trained Models

This 2022 paper analyzes the generated lyrics of a pre-trained GPT-2 model. They elect to use intrinsic perplexity to measure the semantic cohesion of the generated text. The paper concludes that the while models generated semantically and syntactically correct lyrics. While this paper emphasizes the analysis of lyrics, my project will be more focused on generating different types of lyrics with different datasets.

5 Methodology

5.1 Web Scraping

An AI model is only as good as its training data. Due to the complexity and large number of parameters in transformer models, it is often the case that having more data is always better than having less data. In order to customize the dataset, this project attempted to download the all the lyrics from the entire discography of every artist from IMDB's 50 Best Males Rappers of all Time. The motivation for this is that the 'best' rappers should have the highest quality lyrics, which should help the model produce high quality lyrics. The top 50 site is scraped to produce a list of 50 rappers in python.

Lyric scraping was done using Python's BeautifulSoup on https://www.azlyrics.com. This website is ideal because its links to artists' are easy to access. The process for scraping all the lyrics involves two steps: gathering all the links to the lyric pages and gathering the lyrics from the previously generated list of links.

Fortunately, the link to every artist page is algorithmically generated and straightforward to mimic using just the artist name.

Artist Name	Link to Artist Page
Drake	https://www.azlyrics.com/d/drake.html
2Pac	https://www.azlyrics.com/19/2pac.html

Figure 1: Converting Artist Name to Link

Once the links to each artist page has been generated and verified, all the song links within the artist page can be collected and stored in a file. After this process was complete, 11,205 lyric links were generated for 44 different artists.

The next step in the process became a considerable bottleneck in the progress of this project. Unfortunately, established websites such as azlyrics.com have bot detection algorithms to prevent people from sending an excessive number of network requests. The workaround is to wait around 12 seconds between each request. This made the data collection process very time-consuming (would have taken around 48 hours of consecutive runtime). In addition, about halfway through this process, the bot got detected, causing its IP to be banned from future network requests.



Figure 2: Banned from more Network Requests

Eager to continue to ther parts of the project and unwilling to wait for the ban to be lifted, the rest of the artists were scrapped. The final dataset includes 22 different artists and 5842 songs, as shown below in 3.

5.2 Data Cleaning

There are several important steps to convert the data from a raw lyric string to a data structure that is suitable for training a transformer model. First, the data is randomly split into two datasets: a train and validation. Train is used to train the model while validation is used to evaluate the model. A small split size of 0.05 was used to maximize the amount of data used to train the model. Afterwards, using the prebuilt tokenizer, the data is converted from text to a list of tokens which are represented

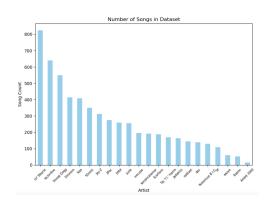


Figure 3: Number of Songs Per Artist in Dataset

by integers and an initialized attention mask. The final step is to reorganize the data into blocks of size 256. This is a standardization method that prevents issues with song lyrics that are too long or short for the gpt model. The main issue here is that all songs are combined and reorganized into blocks, meaning many blocks will really be a combination of two different songs. Fortunately, the dataset is organized by artist and these combined songs will often be from the same artist, and even the same album. After being organized into blocks, the training dataset now has 18,967 rows.

5.3 Model Training

Training involves using the processed dataset to fine-tune the distilgpt2 model. This is a 'distilled' version of GPT-2 (hence the name), making it a solid lightweight option for text generation tasks. The training parameters are set to the defaults recommended in the model's documentation.

```
model_name='distilgpt2'

model_name_new = model_name.split("/")[-1]
training_args = TrainingArguments(
    f"{model_name_new}-rhyme-bot-1.0",
    evaluation_strategy = "epoch",
    save_strategy = 'epoch',
    load_best_model_at_end = True,
    num_train_epochs=10,
    learning_rate=2e=5,
    weight_decay=0.01,
    warmup_steps = 100.0,
)
```

Figure 4: Parameters for Distilgpt2 Model Training

It is integral that the model training takes place in a Cuda-compatible environment such as Google Colab. This reduces the training time from over 72 hours to just about 3. Once the model has finished training it can be saved and loaded for future use.

It is worth noting that the GPU requirement for

model training led to a number of headaches and significantly stunted the capabilities of the model. Unfortunately, the free version of Google Colab offers limited GPU usage. This led to several models to shutdown after hours of training due surpassing Google's GPU limit. To comply with this rule, the model could only be trained on 10 epochs, despite it still improving at this point.

5.4 Model Evaluation

The evaluation of a transformer model is the point of much debate in the natural language processing community. Ultimately, the choice of evaluation metrics should depend on the specific task that the model is performing. In lyric generation, the quality of lyrics is highly subjective, causing the formulation of an objective metric to be difficult. For this reason, the negative log-likelihood is used for evaluation in the training of the model, and subjective human evaluation is used to evaluate the model once it has been trained.

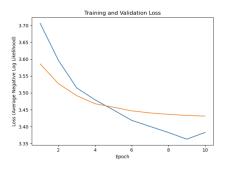


Figure 5: Loss at each Epoch

No formal process was used for subjective evaluation. Instead lyrics were inspected and evaluated on whether they made sense in the context of hip hop and also were able to logically continue the prompt lyrics.

6 Conclusion

Large transformer models such as Chat-GPT may soon (if they have not already) deem traditional natural language processing techniques obsolete. Learning to work with these powerful models and understanding their strengths and limitations is essential for advancing in the field of natural language processing. This project has demonstrated a feasible and reproducible approach to fine-tuning transformer models for specific tasks. By leveraging web scraping techniques, data preprocessing, and model training, Rhymebot 1.0 was successfully developed and trained on a curated dataset of hip-hop

lyrics. Despite challenges such as bot detection algorithms and GPU limitations, the project achieved its primary objective of creating a functional AI model capable of generating and auto-completing hip-hop lyrics.

219

220

221

227

237

239

241

242

243

245

247

251

260261

263

264

265

267

There are a myriad of ways this model can be improved. Increasing the size of the dataset is an obvious one. Spending \$10 for a month of Colab Pro and increased GPU so that the model can be trained for longer is another. Additionally, due to the GPU limitations and long training times, experimenting with different parameter configurations was not feasible. This could have led to marginal improvements. A point worth candidly mentioning is that not much time was left to spend evaluating the model after it was built. Designing a framework for evaluating the model using different prompts and metrics would have been a great addition to the project. While the 'songs' in the appendix pass the eye test, this is hardly a scientific method for evaluating a text generation model, and the evaluation section is the weakest of this report.

In conclusion, the development of Rhymebot 1.0 demonstrates A.I.'s ability to be useful in creative endeavors such as writing lyrics. While AI-generated lyrics may never fully replace the creativity and authenticity of human expression, they can serve as a valuable tool for inspiring and augmenting artistic creation. As technology continues to evolve, the boundaries between human and AI creativity will blur, opening up new possibilities for collaboration and innovation in the world of music and beyond.

7 Appendix

7.1 Song 1

Prompt:

Ayo, y'all other rap bots got no game I drip hard, don't need the fame I got the training data to back me up Your weak rhymes, they're just a hiccup Your data's shallow, can't stand the heat While I spit fire, you're just obsolete

Model Output:

Ayo, y'all other rap bots got no game I drip hard, don't need the fame I got the training data to back me up Your weak rhymes, they're just a hiccup

Your data's shallow, can't stand the heat

While I spit fire, you're just obsolete	268
Your data's no longer safe for the rest of yo	269
With the same intentions and I'm a dog in this bito	b7on t
	271
You're a hypocrite then I'm a sucker	272
for the bullshit	273
I'm a motherfucker with the shit	274
that I put on my head, man	275
You better watch your mouth,	276
man, watch your face	277
You better watch your teeth,	278
man, watch your mouth	279
You don't ever see me in my life	280
Look me in your eyes, man	281
Look me in your eyes, man	282
I got plans, for you to get on the set with me	283
You got plans, for you to get	284
7.2 Song 2	285
Prompt:	286
-	
We out here in Boulder	287
Reppin hard that rocky mountain kush	288
	289
Model Output:	290
We out here in Boulder	004
Reppin hard that rocky mountain kush	291
Reppin hard that rocky modificati kush	292293
You wanna take me in if you look at me	293
What you wanna take me	295
Caught up with this thug gangsta shit	296
caught up with this thug gangsta shit	297
From the East to West Coast	298
From Cali to L.A.	299
Rappers know how we ride to a high end	300
rapper o fator from the Frace to a first cha	301
From Cali to L.A.	302
Rappers know how we ride to a high end	303
Nappor o Tillon Hon Ho I Zuo do di Hagir ella	304
Ay, what you wanna do?	305
It's all about where you're at	306
Rappers know where you're at	307
Rappers know where you're at,	308
It's all about where you're at	309
Rappers know where you're at	310
Rappers know where you're at	311
I got that gangsta shit I'm bustin on	312

We gon' bust 'em, we gon' bust 'em

Prompt:

313

314

315

7.3 Song 3

In the city where the streets don't sleep, where every corner holds a story deep. From broken dreams to shattered hopes, this concrete jungle's where I cope.

Model Output:

In the city where the streets don't sleep, where every corner holds a story deep. From broken dreams to shattered hopes, this concrete jungle's where I cope. What's the difference?

I know everybody's happy 'fore I die, but I know I'm wrong
I know it's sad how the system do
I know my mama's gone
She's always there for the wrong reasons
Still my baby loves me, but I only know
I'm not there for me
Where my baby sees me, we do the same thing
In the darkness, she feels my pain
(It's a shame)

Why?
Why did I feel so damn lost?
(It's a shame)

What do I do for my people?
Let me know what we can do for all of these people
We're going all out here, for all of these people
We're going all out here

7.4 Song 4

Prompt:

That's why I fucked yo' bitch, you fat motherfucker West side, Bad Boy killers get 'em, and I'mma shoot em

Model Output:

That's why I fucked yo' bitch, you fat motherfucker West side, Bad Boy killers get 'em, and I'mma shoot em

I'mma shoot the motherfuckin' crip And I'mma murder crip, in my hand, hee-ho Got him a nine-five, but he like to shoot him

Them dudes like to shoot the motherfucker, you fat motherfucker

Them guns, I got money, so I'mma shoot him I'mma shoot 'em, 'cause your boy shoot 'em Can't even see him, so what up, let's get away I'mma shoot him, I'mma shoot 'em

I wanna see you, but, can't believe I can see you See I can't see you, but, can't believe I can see you I got money, so I'mma shoot 'em Can't

Works Cited

- 1. Potash, Peter, Alexey Romanov, and Anna Rumshisky. "Ghostwriter: Using an 1stm for automatic rap lyric generation." Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing. 2015.
- 2. Wu, X., Du, Z., Guo, Y. et al. Hierarchical attention based long short-term memory for Chinese lyric generation. Appl Intell 49, 44–52 (2019). https://doi.org/10.1007/s10489-018-1206-2
- 3. Zhe Zhang, Karol Lasocki, Yi Yu, and Atsuhiro Takasu. 2024. Syllable-level lyrics generation from melody exploiting character-level language model. In Findings of the Association for Computational Linguistics: EACL 2024, pages 1336–1346, St. Julian's, Malta. Association for Computational Linguistics.
- 4. Rodrigues, M. A., Oliveira, A., Moreira, A., Possi, M. (2022). Lyrics Generation supported by Pre-trained Models. The International FLAIRS Conference Proceedings, 35. https://doi.org/10.32473/flairs.v35i.13060